## **🕒 Timers in JMeter – Overview**

**What are Timers?** Timers are used to simulate real user behavior by adding **delays** (aka *think time*) between requests.

Without timers, JMeter would send all requests **as fast as possible**, which isn't realistic.

## **🎯 Why Use Timers?**

| **🧩 Purpose** | **📘 Explanation** |
| --- | --- |
| 🧠 Simulate Real User Behavior | Real users pause before actions. Timers replicate these pauses. |
| 🛡 Prevent Server Overload | Avoid sending too many requests too quickly. |
| 🎯 Maintain Desired Throughput | Achieve consistent requests-per-minute/second with timers like *Throughput*. |

## **🧭 How Timers Work**

### **🔄 Scope**

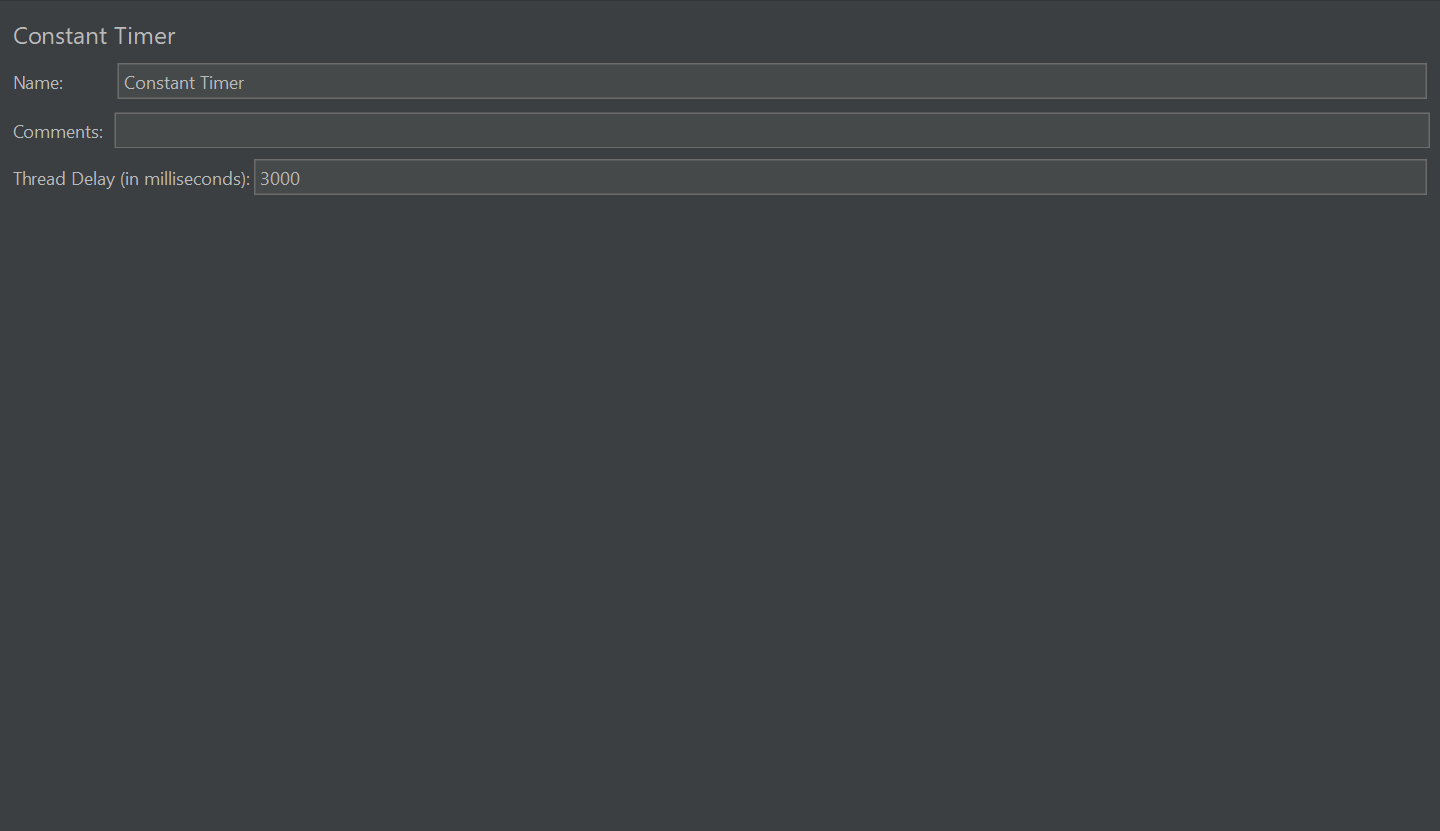
| **Placement Location** | **Effect on Samplers** |
| --- | --- |
| Under Thread Group | Applies to **all samplers** inside that group. |
| As Child of a Sampler | Applies **only to that sampler** before its execution. |

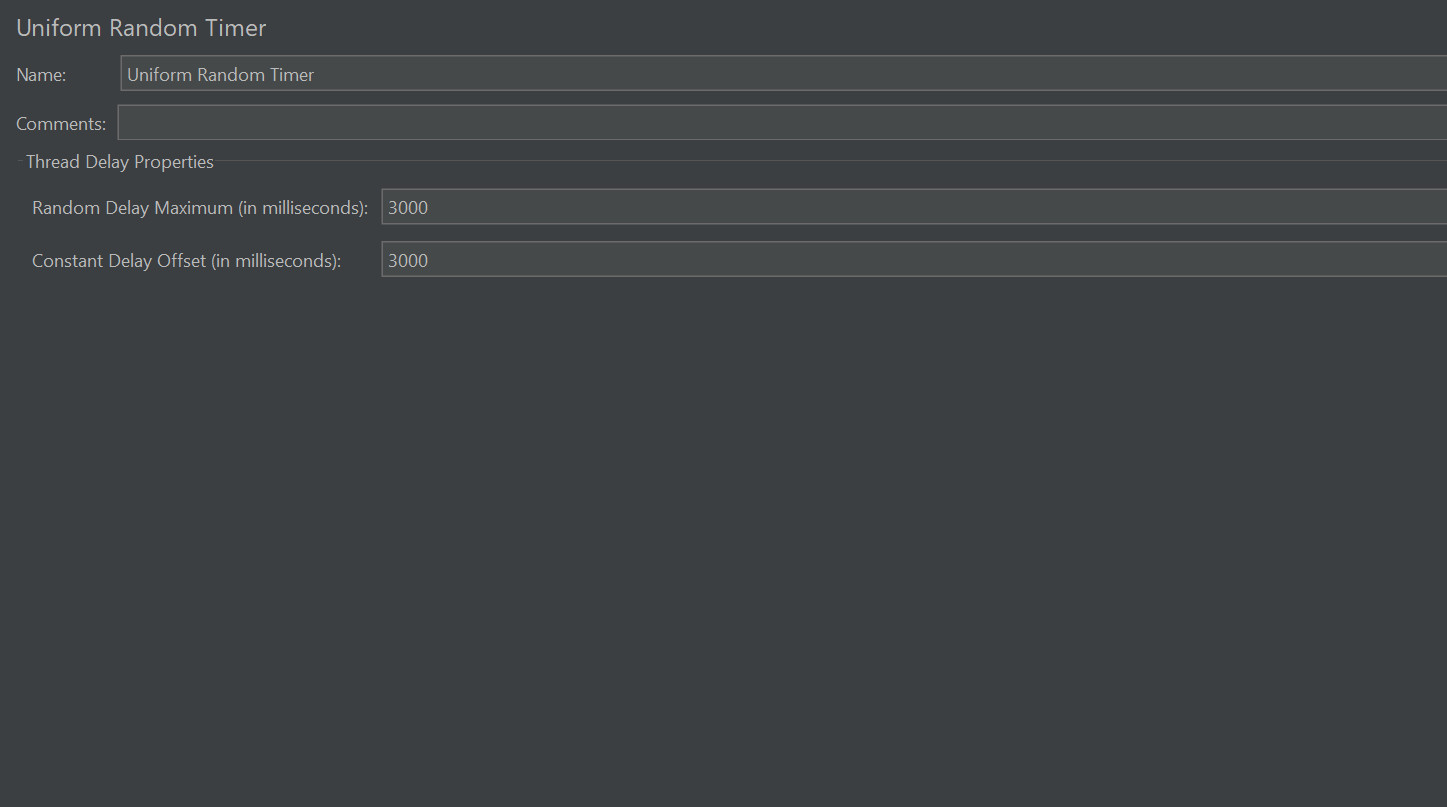
### **🧷 Execution Order**

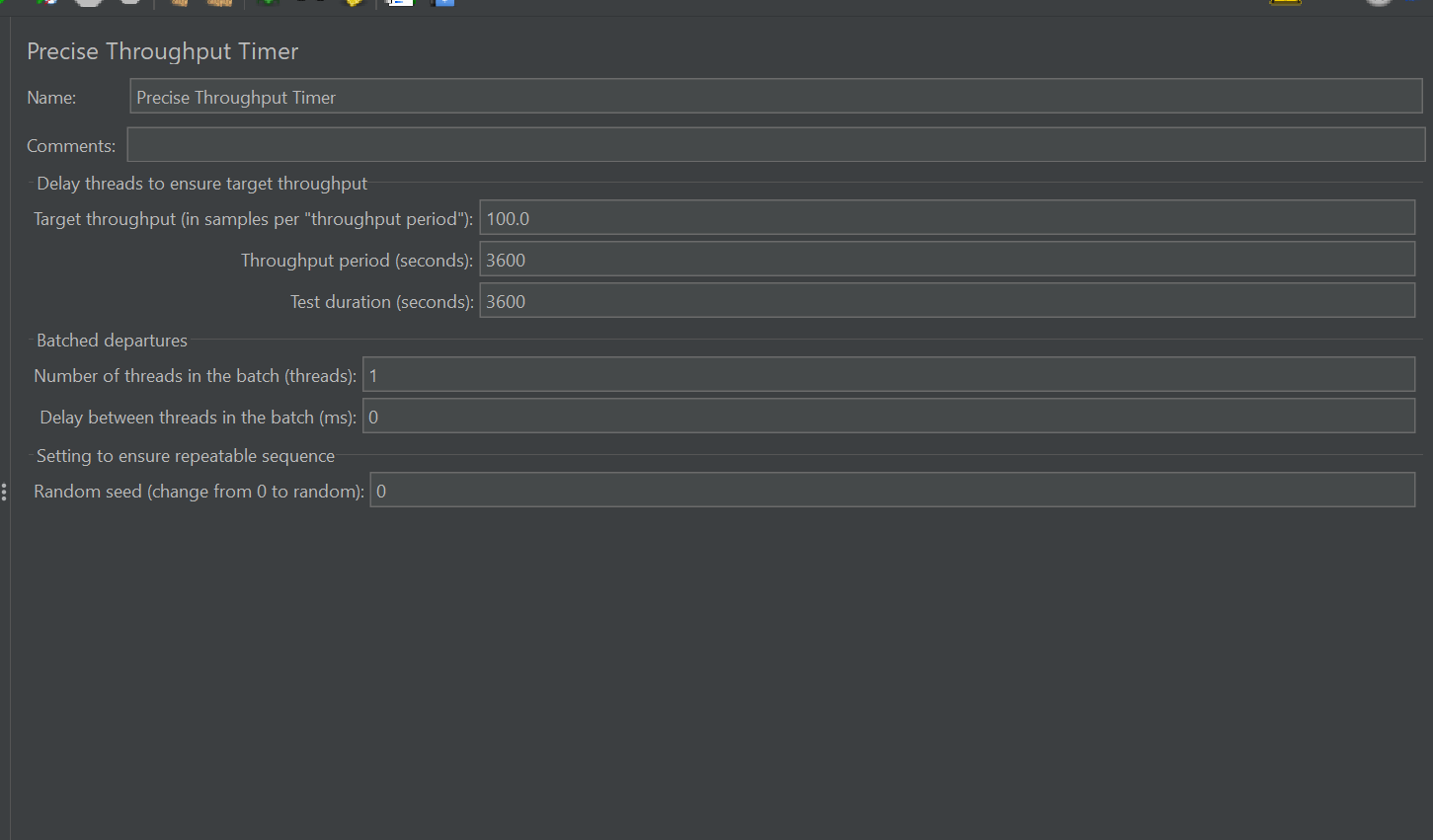
* All timers within scope **execute before the sampler**.
* Multiple timers in scope? ⏱ Their delays are usually **summed** (unless stated otherwise).

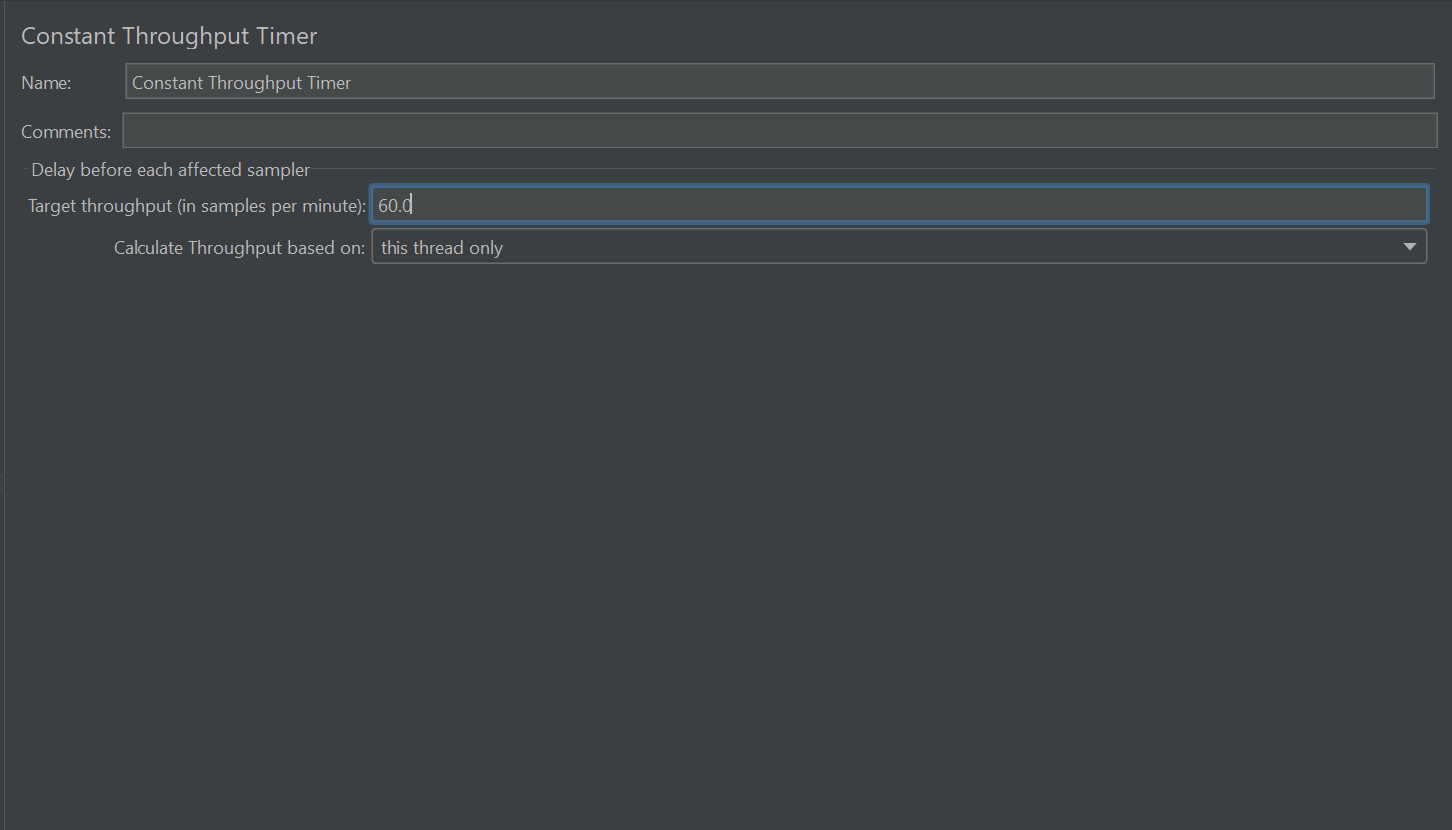
## **⏱ Common Timer Types**

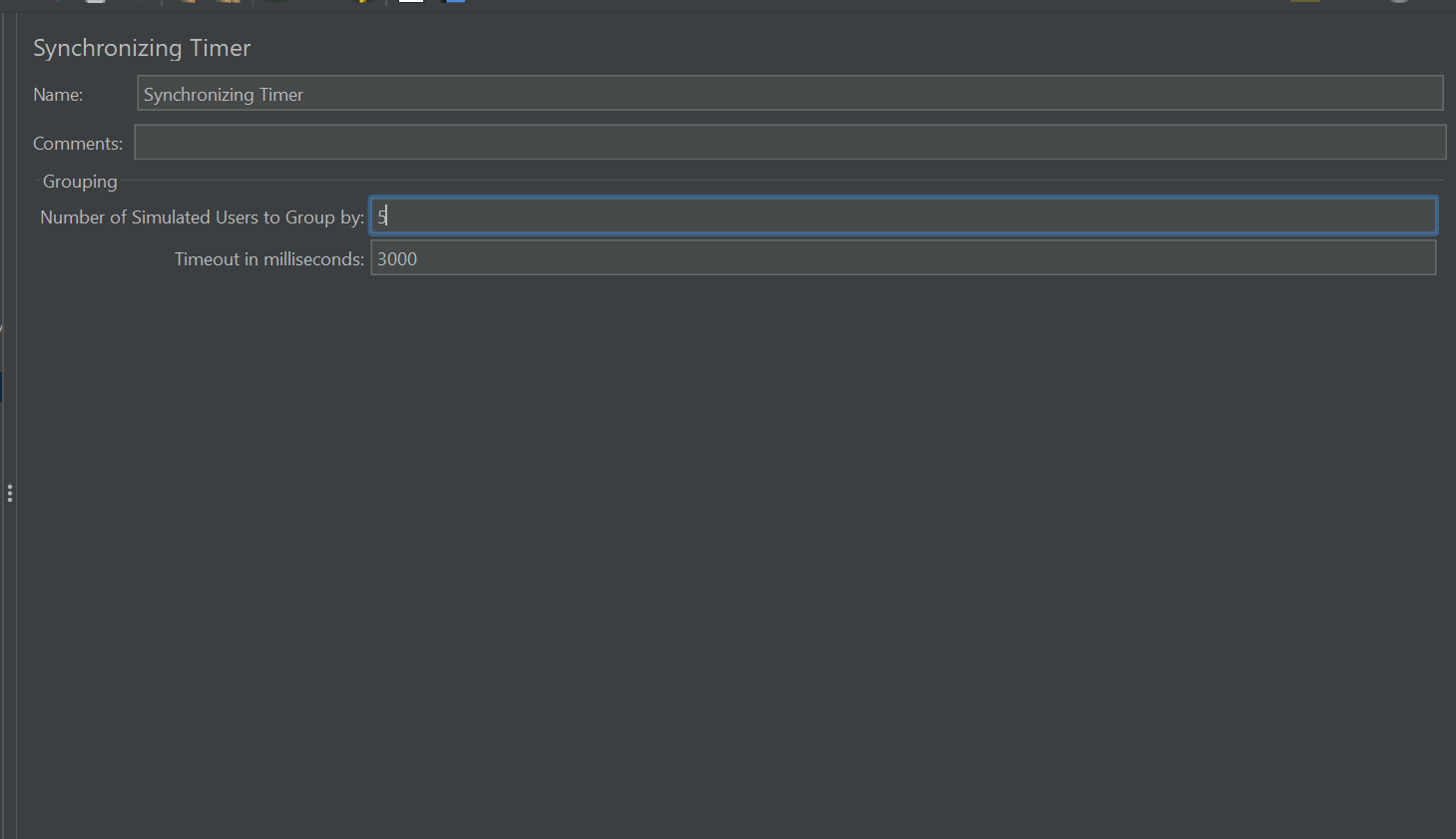
| **Timer Type** | **🔍 Purpose & Use Case** | **⚙️ Configuration Highlights** |
| --- | --- | --- |
| 🧮 **Constant Timer** | Fixed delay between requests.  🔧 *Use for consistent delays.* | Delay (ms) |
| 🎲 **Uniform Random Timer** | Adds randomness to think time.  🔧 *Good for varying user behavior.* | Max Delay, Offset |
| 📊 **Gaussian Random Timer** | Realistic human-like distribution of delays. | Deviation, Offset |
| 📈 **Poisson Random Timer** | Events based on average arrival rate.  🔧 *Great for traffic simulation.* | Lambda, Offset |
| 📏 **Constant Throughput** | Maintains exact number of requests per minute.  🔧 *Use to simulate load profile.* | Target Rate, Scope Type |
| ⏳ **Synchronizing Timer** | Syncs threads to release them **all at once**.  🔧 *Use for spike testing.* | Thread Count, Timeout |
| 💻 **JSR223 / BeanShell** | Custom scripting for delays.  🔧 *For advanced/custom delay logic.* | Script in Groovy, JS, etc. |











## **📌 How to Add a Timer in JMeter**

1. **Right-click** on element (Thread Group, Controller, or Sampler)
2. Choose ➕ Add > Timer
3. Select desired Timer (e.g., Constant Timer)

## **🧠 Best Practices**

| **✅ Practice** | **💡 Tip** |
| --- | --- |
| 👥 Mimic Real User Behavior | Base your delays on analytics or logs (how long users actually take). |
| 📍 Understand Scope | Timer placement impacts where it applies — avoid unintended global delays. |
| 🚫 Avoid Heavy Listeners During Load | Disable "View Results Tree" or similar during large tests. |
| 📶 Combine with Ramp-Up | Use ramp-up with timers to simulate gradual traffic growth. |
| 📈 Monitor Server Metrics | Timers simulate clients — but always watch server CPU, memory, disk, etc., during tests. |

## **📚 Example: Constant Timer**

jmeter

CopyEdit

Thread Group (10 users, Ramp-up: 10 sec)

├── HTTP Request (GET /login)

│ └── Constant Timer (3000 ms)

├── HTTP Request (POST /submit-form)

📝 This setup adds a 3-second delay before /login request only.